



**PUC Minas**  
DIRETORIA DE  
EDUCAÇÃO CONTINUADA

Pós Graduação *Lato Sensu*

Bancos de dados não  
relacionais

Operações Básicas

# Operações básicas

Utilizando MongoDB vamos realizar operações de:

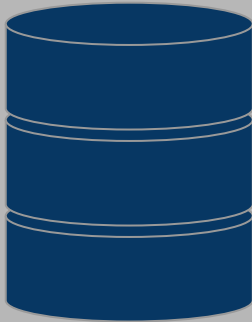
- Inserção
- Pesquisa
- Atualização
- Remoção

# Porque mongoDB?

- Documentação
- Comunidade
  - Google groups:
    - mongodb-user
    - br-mongodb
  - IRC
    - server: irc.ubuntu.com
    - room: #mongodb
- Utilizado por várias empresas

# Mongo

mongod



mongo

```
> show dbs
```

```
test  
puc
```

```
> use puc  
> show collections
```

```
Alunos
```



# NoSQL Documento

```
{  
  _id: ObjectId("507f191e810c19729de860ea")           // Qualquer tipo de dado  
  name: "Nome",  
  age: 25,  
  birthday: ISODate("2016-11-10T19:13:00Z"),  
  notes: [5,5,5,5],  
  informations: {  
    city: "Belo Horizonte"  
    state: "Minas Gerais"  
  },  
  friends: [{  
    "name": "Nome 2"  
  }]  
}
```

# Inserção

use aula2

```
db.Alunos.insert({  
  "name": "Nome",  
  "birthday": $gte: ISODate("2016-11-10T19:13:00Z"),  
  "group": "Grupo 1"  
})
```

# Pesquisa

```
db.Alunos.find({  
  "name": "Nome",  
})
```

```
db.Alunos.find({  
  "name": /me/,  
})
```

```
db.Alunos.find({  
  "birthday": {  
    $gte: ISODate("2000-01-01T00:00:00Z")  
  }  
})
```

## Operadores

Comparadores: \$gt, \$gte, \$lt, \$lte, \$ne

Array: \$in, \$nin, \$all, \$size

Lógicos: \$or, **\$and**, \$not, \$nor

Elementos: **\$exists**, \$type

Modulares: \$mod, \$regex, \$text, \$where

Geoespaciais: \$geoWithin, \$geoIntersects, \$near

# Atualização

```
db.Alunos.update(  
  {  
    "name": "Nome"  
  },  
  {  
    $set: {  
      "name": "Gabriel",  
      "note": 10  
    }  
  }  
)
```

## Operadores

\$inc, \$mul, \$rename, \$setOnInsert, \$set, \$unset,  
\$min, \$max, \$currentDate

\$addToSet, \$pop, \$pullAll, \$pull, \$pushAll,  
\$push

\$each, \$slice, \$sort, \$position



# Apagar

```
db.Alunos.delete({  
  "nome": "Gabriel"  
})
```

```
db.Alunos.delete({  
  "name": /el/,  
})
```

# Importando e exportando dados

## Exportar

### **mongodump:**

Exporta toda a estrutura do mongo incluindo índices.  
Escreve dentro de uma pasta específica chamada dump

### **mongoexport:**

Exporta de um determinado banco e coleção o documento em formato JSON

## Importar

### **mongorestore:**

Importa o dump de um outro banco de dados

### **mongoimport:**

Importa para um determinado banco e coleção o documento em formato JSON

# Dúvida nos trabalhos

## Seminário

Dia: 01/12/2016

Encontre algum caso de sucesso ou insucesso e apresente para a classe os principais pontos desses casos.

Grupos de 4 ou 5 pessoas

Máximo 20 minutos

## Trabalho Prático

Dia: 16/12/2016

Coletar informações de redes sociais e armazenar no mínimo 1M de dados.

Individual

# Seminários

**Grupo 01** - O MongoDB aplicado ao CartolaFC - Amanda, Marcelo Oliveira, Marcus e Sérgio

**Grupo 02** - Ebay utilizando NEO4J - Jucélio, Thiago, Fernando, José e Viviane

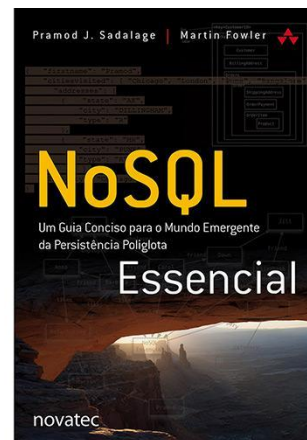
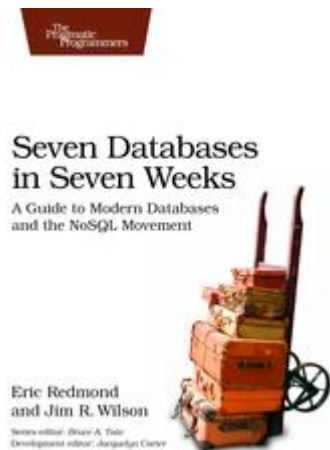
**Grupo 03** - Leroy Merlin - Willian, João, Hernani e Breno

**Grupo 04** - CASE Telefônica ou Bosch - Eduardo, Leonardo, Cristiano, Fabricio, Briner

**Grupo 05** - NoSQL na Netflix - Marcelo, Alan, Robert, Gilmar

**Grupo 06** - Um estudo sobre a utilização do banco de dados NoSQL Cassandra em dados biológicos / biodiversidade - Pedro, Francisco, Eduardo Augusto e Marcelo dos Santos.

# Referências



# Ligando o mongo

Abrir o terminal  
cd ~/Aulas/mongodb

**mongod**

```
./bin/mongod --dbpath=../dados
```

**mongo shell**

```
./bin/mongo
```

# Exercício 1

- Escolha 3 colegas e:
  - Insira no banco informações sobre você e seus colegas como nome, data de nascimento disciplinas cursadas e em curso na PUC
  - Procure no banco a pessoa com a menor data de nascimento
- Atualize a sua nota na disciplina NoSQL para 5
- Apague um de seus colegas
- Copie e cole essas operações em um arquivo e adicione em seu github em uma pasta chamada aula2

# MongoDB & Python

```
import pymongo
```

```
if __name__ == "__main__":
```

```
    client = pymongo.MongoClient("localhost")
```

```
    db = client.nosqlclass
```

```
    var cursor = db.Vocabulary.find({})
```

```
    for document in cursor:
```

```
        print(document)
```



# Exercício 2

## Contando palavras

Primeiramente importe os dados do arquivo `~Aulas/nosql-class/aula2/Vocabulary.json` para o banco de dados `nosqlclass` e a coleção `Vocabulary`

```
./bin/mongoimport -d nosqlclass -c Vocabulary nosql-class/aula2/Vocabulary.json
```

Complete o código do arquivo que está dentro de `~Aulas/nosql-class/aula2/ex2.py`

- Liste as dez palavras com o maior valor no campos `total` e as imprima na tela.
- Encontre todas as palavras que são usuários do twitter, hashtags, urls e adicione um campo informando o tipo da palavra.
- Conte o total de cada um dos tipos que você criou.

# Exercício 3

Sistema de cadastro de alunos:

- O sistema deve permitir inserir um novo aluno com nome, data de nascimento
- Adicionar a um aluno disciplinas que está cursando e cursadas
- Adicionar notas de um aluno a uma determinada disciplina
- Adicione esse arquivo em seu github em uma pasta chamada aula2